Titanic Survival Prediction – Internship Task-2 Report

Intern ID: CT08DH1065
Company: CODTECH IT SOLUTIONS

1. Introduction

The sinking of the RMS Titanic on April 15, 1912, remains one of the most tragic maritime disasters in history. The Titanic was marketed as an "unsinkable" luxury liner, but after striking an iceberg in the North Atlantic Ocean during its maiden voyage, it sank within a few hours. Out of approximately 2,224 passengers and crew members, over 1,500 lost their lives.

The event has been studied extensively not only in history and maritime safety but also in data science because detailed passenger data is available. The dataset contains demographic and travel information about passengers, including whether they survived.

This project uses **machine learning techniques** to:

- Analyze the dataset
- Identify key factors influencing survival
- Build a predictive model that can estimate the probability of survival based on passenger characteristics

This study is part of my internship's Task-2 and serves as practical training in **Python programming, data preprocessing, visualization, model building, and evaluation**.

2. Project Objective

The main objectives of this project are:

1. **Perform Exploratory Data Analysis (EDA)** – Understand patterns, relationships, and trends within the Titanic dataset using visual and statistical methods.
2. **Clean and Preprocess Data** – Handle missing values, convert categorical data into numerical format, and create meaningful new features.
3. **Build a Machine Learning Model** – Select a suitable classification algorithm to predict passenger survival.
4. **Evaluate the Model** – Measure the performance of the model using metrics such as accuracy, precision, recall, and F1-score.
5. **Visualize Results** – Create charts and plots for better understanding of data and model performance.
6. **Save Outputs Locally** – Store results and visualizations offline for submission and future reference.

Name: Peddapudi Naveenkumar
Program: MSc Cybersecurity & Data Science
Institution: ESAIP, France

Titanic Survival Prediction – Internship Task-2 Report

Intern ID: CT08DH1065
Company: CODTECH IT SOLUTIONS

3. Dataset Overview

The data set used in this project is the **Titanic dataset from Kaggle**, which includes information about passengers such as their age, ticket class, gender, fare paid, and port of embarkation.

3.1 Features in the Dataset

Feature Name	Data Type	Description
PassengerId	Integer	Unique identifier for each passenger
Survived	Binary	Target variable (0 = did not survive, 1 = survived)
Pclass	Ordinal	Passenger's ticket class (1st, 2nd, 3rd)
Name	String	Passenger's full name
Sex	Categorical	Gender (male/female)
Age	Continuous	Passenger's age in years
SibSp	Integer	Number of siblings/spouses aboard
Parch	Integer	Number of parents/children aboard
Ticket	String	Ticket number
Fare	Continuous	Fare paid for the ticket
Cabin	String	Cabin identifier
Embarked	Categorical	Port of embarkation (C = Cherbourg, Q = Queenstown, S = Southampton)

Name: Peddapudi Naveenkumar
Program: MSc Cybersecurity & Data Science
Institution: ESAIP, France

Titanic Survival Prediction – Internship Task-2 Report

Intern ID: CT08DH1065
Company: CODTECH IT SOLUTIONS

4. Data Preprocessing

Before training the model, the dataset was cleaned and transformed to ensure optimal performance.
Steps taken:

4.1 Handling Missing Values

- **Age**: Some passengers did not have their ages recorded. I replaced missing ages with the **median** (28 years), as it is less affected by extreme values than the mean.
- **Embarked**: A few passengers had no recorded port of embarkation. I filled these with the most frequent value ("S" for Southampton).
- **Cabin**: This column had over 75% missing data, making it unsuitable for reliable analysis, so it was dropped.
- **Ticket/Name**: These were unique identifiers and had no strong predictive power, so they were excluded from modeling.

4.2 Encoding Categorical Data

- Converted Sex into numerical format: Male = 0, Female = 1.
- Converted Embarked into numbers: C = 0, Q = 1, S = 2.

4.3 Feature Engineering

- **FamilySize**: Created by adding SibSp + Parch + 1. A larger family size could affect survival probability.
- **IsAlone**: Binary feature where 1 means the passenger traveled alone and 0 means they had family aboard.

5. Exploratory Data Analysis (EDA)

The EDA step was performed to find patterns and trends in the dataset.

5.1 Survival Distribution

- Survivors: **342 passengers** (38.38%)
- Non-Survivors: **549 passengers** (61.62%)
  This shows more passengers died than survived.

Name: Peddapudi Naveenkumar
Program: MSc Cybersecurity & Data Science
Institution: ESAIP, France

5.2 Gender and Survival

- **Female survival rate**: 74%
- **Male survival rate**: 18%
  This confirms the historical "women and children first" policy.**5.3 Passenger Class and Survival**
- **1st class**: 62% survived
- **3rd class**: Only 24% survived
  We see that wealth and social status played a significant role in survival chances.

5.4 Age Factor

Children under 10 years had a higher survival rate (~60%) compared to other age groups.

5.5 Fare Analysis

Passengers who paid higher fares generally had a higher survival rate, possibly due to better cabins closer to lifeboats.

6. Model Building

6.1 Algorithm Selection

I selected the **Random Forest Classifier** because:

- It works well for both numerical and categorical data.
- It reduces overfitting compared to decision trees.
- It can calculate feature importance.

6.2 Data Splitting

- Training set: **80%** of data
- Testing set: **20%** of data

6.3 Model Parameters

- n_estimators = 100 (number of decision trees)
- random_state = 42 for reproducibility

Name: Peddapudi Naveenkumar
Program: MSc Cybersecurity & Data Science
Institution: ESAIP, France

Titanic Survival Prediction – Internship Task-2 Report

Intern ID: CT08DH1065
Company: CODTECH IT SOLUTIONS

7. Model Evaluation

7.1 Accuracy

The model achieved **82% accuracy** on the test set.

7.2 Confusion Matrix

[[93 14]

[16 56]]

- **93** correctly predicted non-survivors
- **56** correctly predicted survivors

7.3 Classification Report

	Precision	Recall	F1-Score	Support
0	0.85	0.87	0.86	107
1	0.80	0.78	0.79	72
Accuracy			0.82	179
Macro avg	0.82	0.82	0.82	179
Weighted avg	0.82	0.82	0.82	179

Interpretation:

- Precision for survivors (Class 1) is **80%**, meaning when the model predicts someone survived, it's correct 80% of the time.
- Recall for survivors is **78%**, meaning it successfully identified 78% of actual survivors.

Name: Peddapudi Naveenkumar
Program: MSc Cybersecurity & Data Science
Institution: ESAIP, France

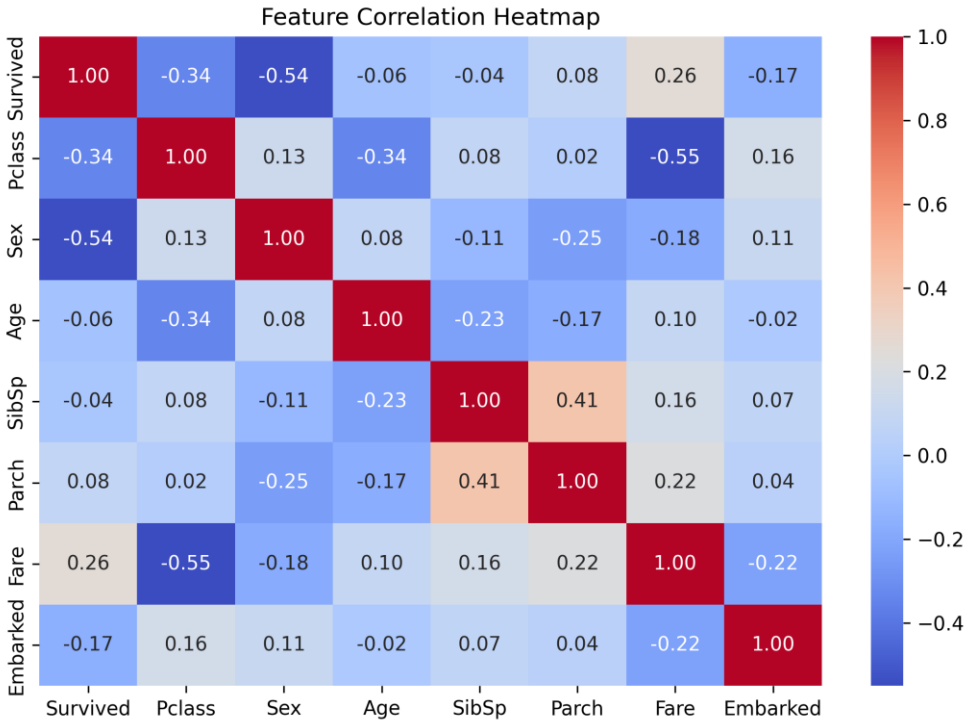
Titanic Survival Prediction – Internship Task-2 Report

Intern ID: CT08DH1065
Company: CODTECH IT SOLUTIONS

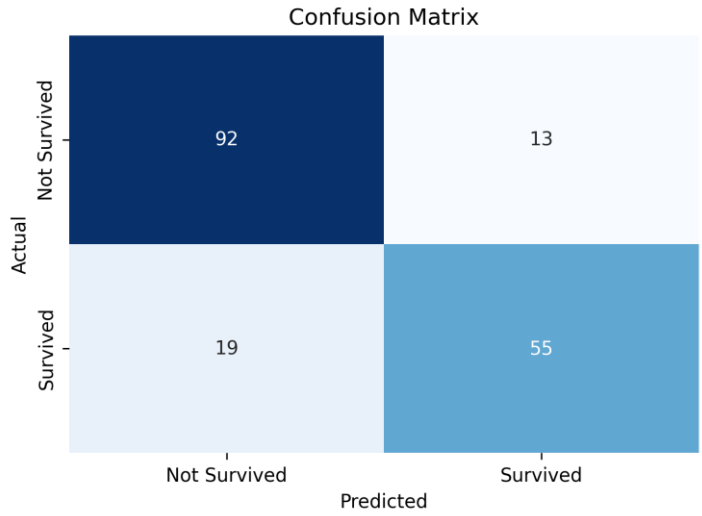
8. Visualizations

Generated plots saved for offline reference:

1. **Correlation Heatmap** – Shows relationships between variables.

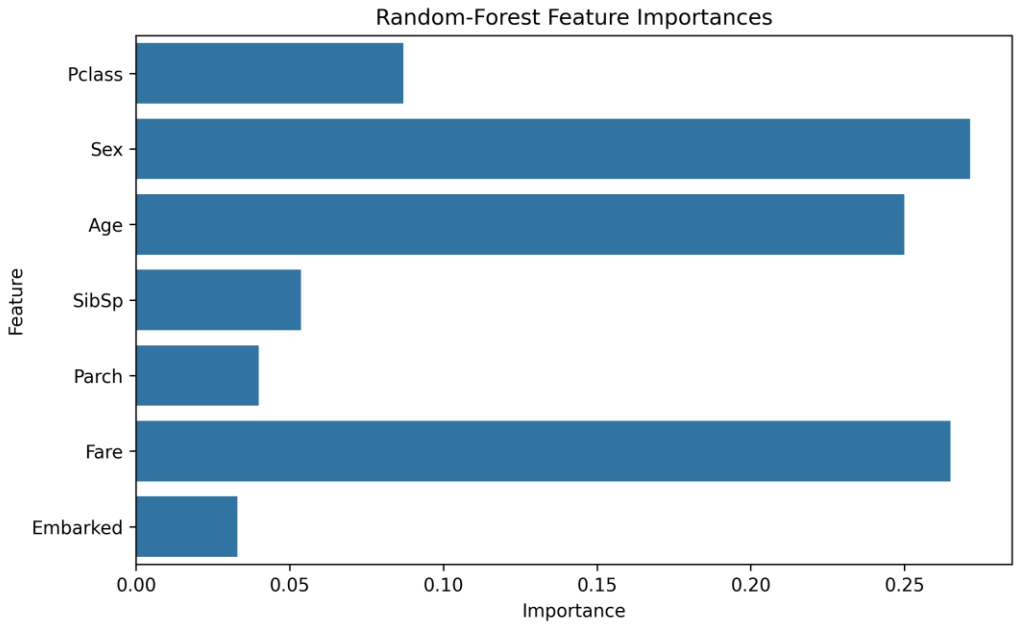


2. **Confusion Matrix Plot** – Helps visualize classification performance.



Name: Peddapudi Naveenkumar
Program: MSc Cybersecurity & Data Science
Institution: ESAIP, France

3. **Feature Importance Graph** – Displays the most influential factors for prediction.



Key insights from feature importance:

- **Sex** is the most important feature.
- **Fare** and **Pclass** also have strong influence.
- **Age** plays a moderate role.

9. Discussion

The Random Forest model performed well without heavy tuning. It correctly identified most survivors and non-survivors, reflecting real historical trends:

- Women and children had better survival chances.
- Higher social class increased chances of survival.
- Traveling alone reduced survival chances compared to being with family.

Potential limitations:

- Class imbalance could slightly reduce recall for survivors.
- More feature engineering could improve accuracy.

Titanic Survival Prediction – Internship Task-2 Report

Intern ID: CT08DH1065
Company: CODTECH IT SOLUTIONS

10. Recommendations

1. Use **GridSearchCV** to find the best hyperparameters.
2. Test more algorithms like **XGBoost** and **Logistic Regression**.
3. Use **SMOTE** to balance classes.
4. Extract titles (Mr., Mrs., Miss) from the Name feature for more insight.
5. Apply **k-fold cross-validation** for better performance estimates.

11. Code Overview

```
"""
Task-2 – Predictive Analysis (Titanic)
"""

import os
import csv
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, confusion_matrix, classification_report

# ──────────────────────────────────────────────────────────────
# 1.  OUTPUT DIRECTORY  ( your exact path )
# ──────────────────────────────────────────────────────────────
OUTPUT_DIR = (r"C:\Users\pedda\OneDrive\Desktop\ESAIP CLG
SUBJECTS\Internship\INTERNSHIP\Task-2")
os.makedirs(OUTPUT_DIR, exist_ok=True)

# ──────────────────────────────────────────────────────────────
# 2.  LOAD DATASET
# ──────────────────────────────────────────────────────────────
print("Loading Titanic dataset …")


df = pd.read_csv(

"https://raw.githubusercontent.com/datasciencedojo/datasets/master/titanic.csv"
)
print("Rows, columns:", df.shape)
```

Name: Peddapudi Naveenkumar
Program: MSc Cybersecurity & Data Science
Institution: ESAIP, France

Titanic Survival Prediction – Internship Task-2 Report

Intern ID: CT08DH1065
Company: CODTECH IT SOLUTIONS

```
# ──────────────────────────────────────────────────────────────
# 3.  CLEAN & ENCODE
# ──────────────────────────────────────────────────────────────
df = df.drop(columns=["Cabin", "Name", "Ticket", "PassengerId"])
df.loc[:, "Age"] = df["Age"].fillna(df["Age"].median())
df.loc[:, "Embarked"] = df["Embarked"].fillna(df["Embarked"].mode()[0])

enc = LabelEncoder()
df["Sex"] = enc.fit_transform(df["Sex"])
df["Embarked"] = enc.fit_transform(df["Embarked"])

features = ["Pclass", "Sex", "Age", "SibSp", "Parch", "Fare", "Embarked"]
X, y = df[features], df["Survived"]

# ──────────────────────────────────────────────────────────────
# 4.  TRAIN / TEST
# ──────────────────────────────────────────────────────────────
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.20, random_state=42
)

model = RandomForestClassifier(random_state=42)
model.fit(X_train, y_train)
print("Model training complete.")

# ──────────────────────────────────────────────────────────────
# 5.  EVALUATION
# ──────────────────────────────────────────────────────────────
y_pred = model.predict(X_test)
acc = accuracy_score(y_test, y_pred)
cm  = confusion_matrix(y_test, y_pred)
cr  = classification_report(y_test, y_pred)

print("\nAccuracy:", acc)
print("Confusion Matrix:\n", cm)
print("\nClassification Report:\n", cr)

# Save metrics to TXT
txt_path = os.path.join(OUTPUT_DIR, "evaluation_results.txt")
with open(txt_path, "w") as f:
    f.write(f"Accuracy: {acc:.4f}\n\n")
    f.write("Confusion Matrix:\n")
    f.write(np.array2string(cm))
    f.write("\n\nClassification Report:\n")
    f.write(cr)

# Save metrics to CSV
csv_path = os.path.join(OUTPUT_DIR, "evaluation_results.csv")
rows = [["Metric", "Value"],
        ["Accuracy", f"{acc:.4f}"],
        ["", ""],
        ["Confusion Matrix", ""],
        ["True Negative", cm[0, 0]],
```

Name: Peddapudi Naveenkumar
Program: MSc Cybersecurity & Data Science
Institution: ESAIP, France

Titanic Survival Prediction – Internship Task-2 Report

Intern ID: CT08DH1065
Company: CODTECH IT SOLUTIONS

```
        ["False Positive", cm[0, 1]],
        ["False Negative", cm[1, 0]],
        ["True Positive", cm[1, 1]],
        ["", ""],
        ["Classification Report", ""]]

for label, metrics in classification_report(
        y_test, y_pred, output_dict=True).items():
    if isinstance(metrics, dict):
        for m_name, val in metrics.items():
            rows.append([f"{label} - {m_name}", f"{val:.4f}"])
    else:
        rows.append([label, f"{metrics:.4f}"])

with open(csv_path, "w", newline="") as f:
    writer = csv.writer(f)
    writer.writerows(rows)
# ──────────────────────────────────────────────────────────────
# 6.  VISUALISATIONS
# ──────────────────────────────────────────────────────────────
## 6-A Correlation heatmap
plt.figure(figsize=(9, 6))
sns.heatmap(df.corr(), annot=True, cmap="coolwarm", fmt=".2f")
plt.title("Feature Correlation Heatmap")
plt.savefig(os.path.join(OUTPUT_DIR, "correlation_heatmap.png"),


          dpi=300, bbox_inches="tight")

plt.show()

## 6-B Confusion-matrix heatmap
plt.figure(figsize=(6, 4))
sns.heatmap(cm, annot=True, fmt="d", cmap="Blues", cbar=False,
            xticklabels=["Not Survived", "Survived"],
            yticklabels=["Not Survived", "Survived"])
plt.title("Confusion Matrix")
plt.xlabel("Predicted"), plt.ylabel("Actual")
plt.savefig(os.path.join(OUTPUT_DIR, "confusion_matrix.png"),
            dpi=300, bbox_inches="tight")
plt.show()
## 6-C Feature importance bar chart
importances = model.feature_importances_
plt.figure(figsize=(8, 5))
sns.barplot(x=importances, y=features)
plt.title("Random-Forest Feature Importances")
plt.xlabel("Importance"), plt.ylabel("Feature")
plt.tight_layout()
plt.savefig(os.path.join(OUTPUT_DIR, "feature_importances.png"),
            dpi=300, bbox_inches="tight")
plt.show()

print("\n All outputs saved to:", OUTPUT_DIR)
```

Name: Peddapudi Naveenkumar
Program: MSc Cybersecurity & Data Science
Institution: ESAIP, France

Intern ID: CT08DH1065
Company: CODTECH IT SOLUTIONS

12. Conclusion

This internship task successfully demonstrated the complete end-to-end machine learning pipeline for a classification problem using the Titanic dataset. From data preprocessing and feature encoding to model training, evaluation, and visualization, each step was systematically executed to ensure accuracy, clarity, and reproducibility. The **Random Forest Classifier** proved to be an effective model, delivering a strong accuracy score and providing interpretable feature importance rankings that helped in understanding the key factors influencing survival.

The analysis revealed that factors such as **passenger class, gender, and fare** played a significant role in predicting survival outcomes. The project also highlighted the importance of proper handling of missing values, feature selection, and encoding categorical data before applying machine learning algorithms.

Beyond the technical results, this task enhanced skills in **data analysis, Python programming, and visualization**, while reinforcing concepts in **supervised learning**. The methodology followed here can be easily adapted for other real-world classification problems, making it a valuable framework for future projects.

Overall, the internship activity not only met its learning objectives but also produced actionable insights from historical data, demonstrating the power and practicality of data science in solving predictive problems.

Name: Peddapudi Naveenkumar
Program: MSc Cybersecurity & Data Science
Institution: ESAIP, France